/\*

\* Piano.c

\*

\* Created: 15/10/2020 6:46:21 PM

\* Author : Rajesh Jasti

\*/

#define *F\_CPU* 8000000UL // CPU frequency set to 8Mhz

#include <avr/io.h> // Libraries for input/output ports

#include <util/delay.h> // Libraries for delay function

#define ledOn (PORTA |=(1<<0)) // Defined led on

#define ledOff (PORTA &=~(1<<0)) // Defined led off

#define buttonPushed (PINA & (1<<1)) // Defined button pushed

#define pianoOn (PORTB |=(1<<4)) // Defined PORTB buzzer on

#define pianoOff (PORTB &=~(1<<4)) // Defined PORTB to buzzer off

#define c5\_freq = 523; // Frequency stored in variable

#define d5\_freq = 587;

#define e5\_freq = 659;

#define f5\_freq = 698;

#define g5\_freq = 784;

#define a5\_freq = 880;

#define b5\_freq = 988;

#define c6\_freq = 1047;

int count;

int main(void)

{

setup();

while (1)

{

if (buttonPushed) // when S9 is pushed

{

ledOn; // LED turns on

\_delay\_ms(2000); // LED turns on for 2 seconds

ledOff; // LED off after 2 seconds

if (count=0) // If count equal to 0 statement is true

{

OCR1B = c6\_freq; // OCR1B output to PORTB frequency to buzzer

pianoOff // buzzer turns off

\_delay\_ms(1000); // buzzer turns off for one second

OCR1B = b5\_freq;

pianoOff

\_delay\_ms(1000);

OCR1B = a5\_freq;

pianoOff

\_delay\_ms(1000);

OCR1B = g5\_freq;

pianoOff

\_delay\_ms(1000);

OCR1B = f5\_freq;

pianoOff

\_delay\_ms(1000);

OCR1B = e5\_freq;

pianoOff

\_delay\_ms(1000);

OCR1B = d5\_freq;

pianoOff

\_delay\_ms(1000);

OCR1B = c5\_freq;

pianoOff

\_delay\_ms(1000);

count=0; // count reset to 0

}

else // if count not equal to 0

{

count++; // count increments

}

}

else // if button not pushed

{

ledOff; // LED off

}

}

}

void setup(void)

{

DDRB = 0b00010000; // Setup a PORTB PIN4 for output

DDRA = 0b00000001; // Setup a PORTA PIN0 FOR output

TCCR1A = 0b00100000; // Channel B BIT 5:4 set to fast CTC mode, BIT 0:1 Set PWM wave form generation mode CTC

TCCR1B = 0b00001010; // Setup to BIT 4:3 wave for generation mode, BIT 0:1:2 Prescaler 8,

}

/\*

System clock pulse = 1/ cpu\_freq, 1/8000000 = 125nS

TCNTO Increment time period = 125nS x 65536-1 = 8.19mS

\*/